Karlsruhe Institute of Technology

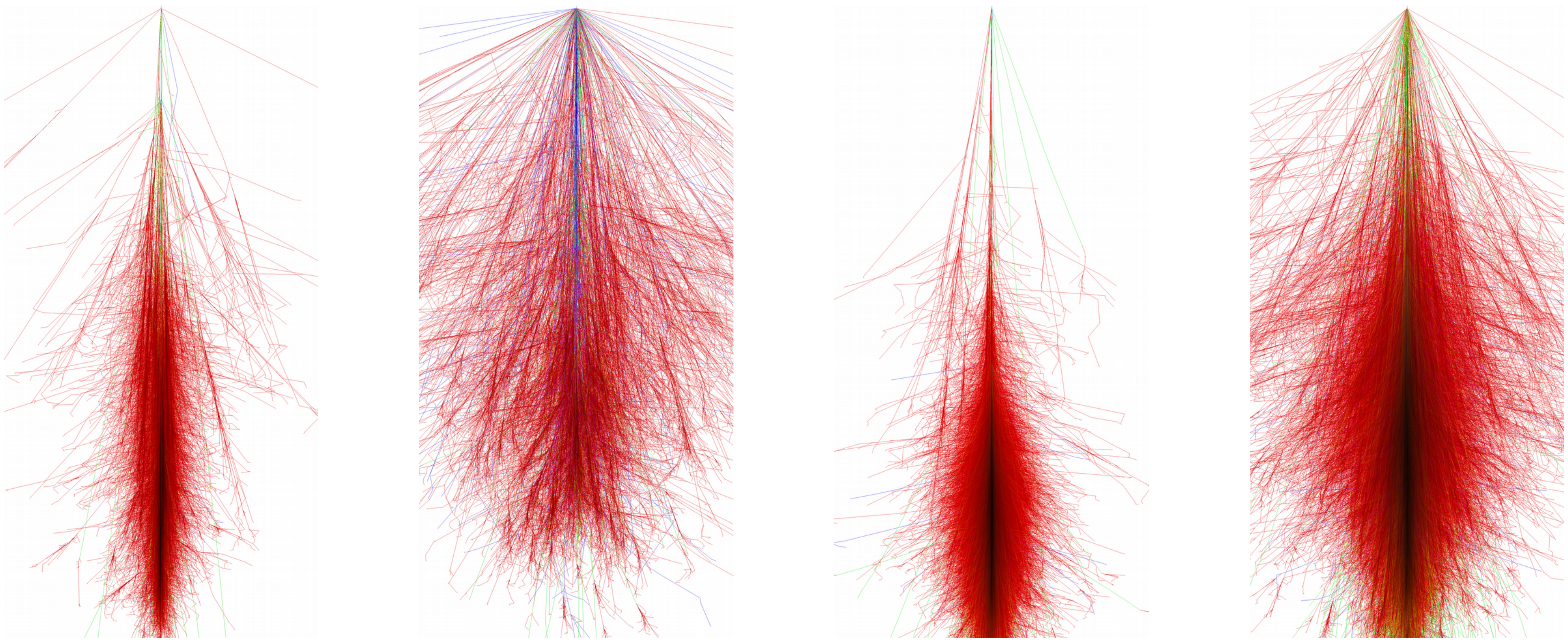
Simulation of Extensive Air Showers with Deep Neural Networks

Marcel Köpke, supervised by Ralph Engel and Markus Roth
HIRSAP Workshop Karlsruhe (2019)

INSTITUTE FOR NUCLEAR PHYSICS (IKP), FACULTY OF PHYSICS
KARLSRUHE INSTITUTE OF TECHNOLOGY (KIT)

CORSIKA 7 [1]

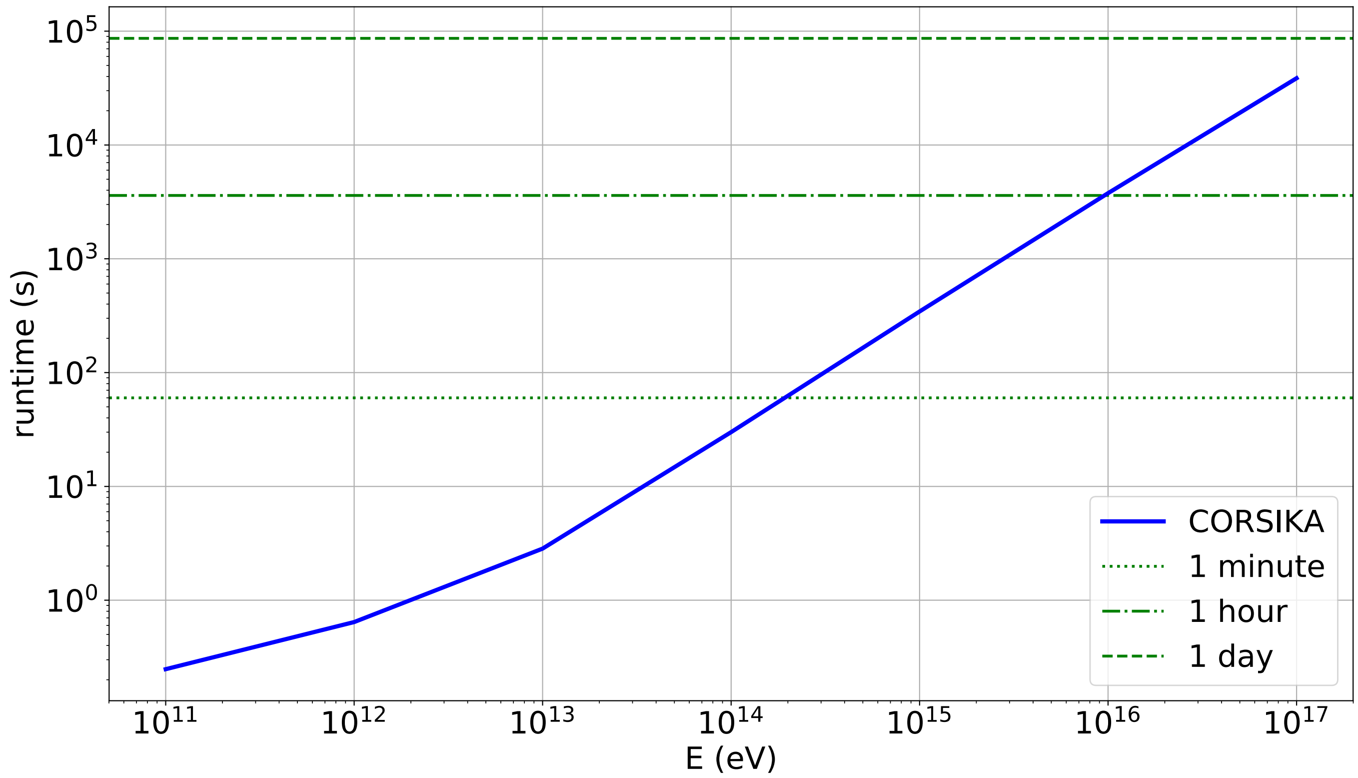
- Extensive air shower Monte Carlo simulation framework
- Different types of interaction models (EPOS-LHC, QGSJET, SIBYLL, ...)



1 TeV Proton | 1 TeV Iron | 10 TeV Proton | 10 TeV Iron

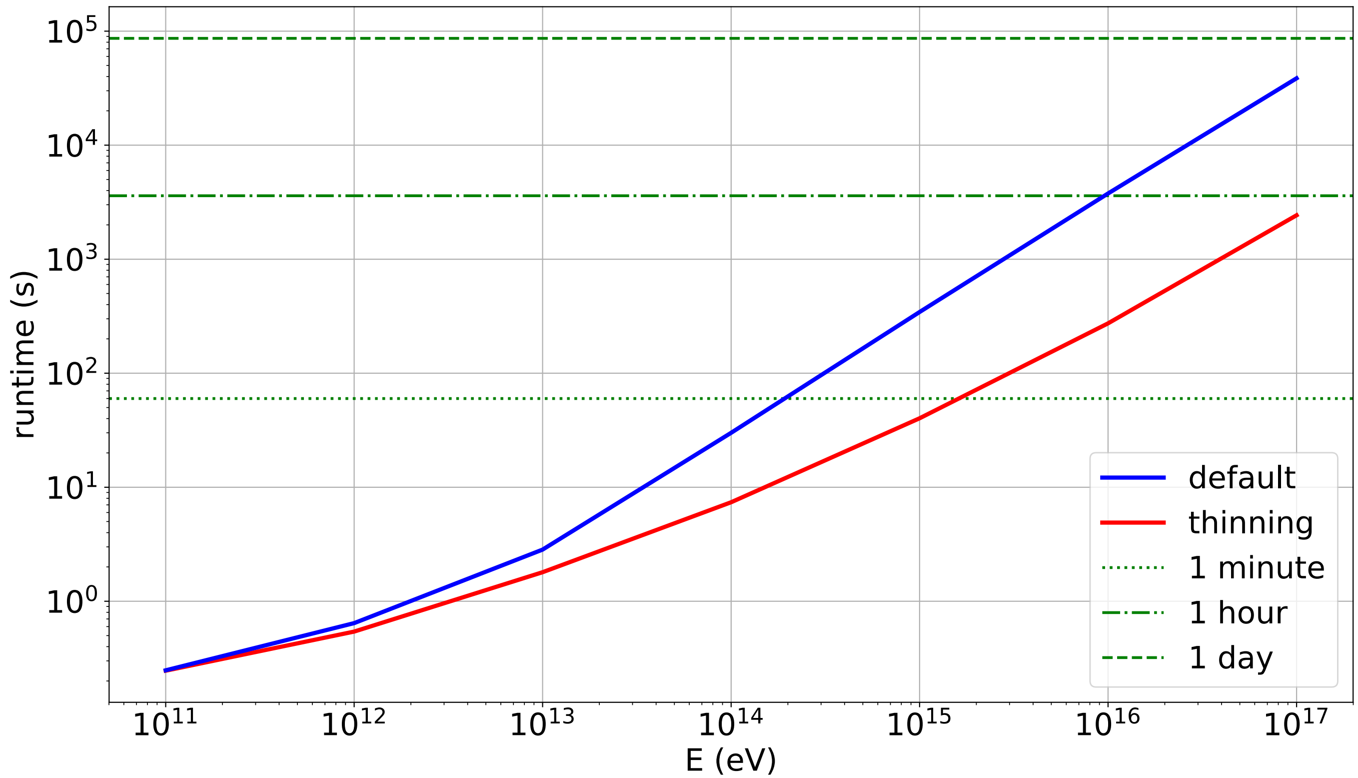
Institute for Nuclear Physics (IKP), Faculty of Physics
Karlsruhe Institute of Technology (KIT)

Motivation

- 

- The time complexity of CORSIKA 7 simulations rises approximately linearly with the primary particle energy

Thinning



- Reduces (effective) particle content by particle-aggregation
- Preserves shower properties to leading order
- Reduces shower-to-shower fluctuations

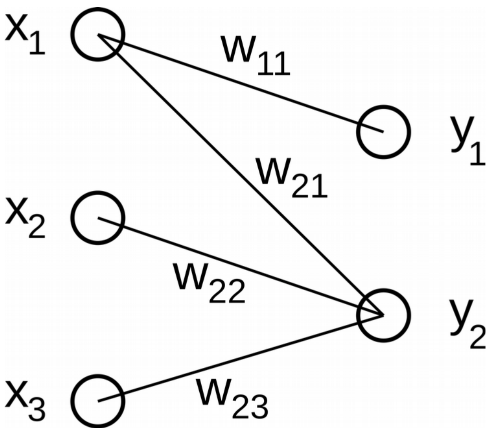
Institute for Nuclear Physics (IKP), Faculty of Physics
Karlsruhe Institute of Technology (KIT)

Why Neural Networks?

Karlsruhe Institute of Technology

- Can run on specialized **hardware** (GPU / TPU)
- Automatic **parallelization** (TensorFlow)
- Automatic reduction to **essential features**
- Training can **fix meta-parameters**
- **Adjustable accuracy** possible

Neural Networks

- 

$$y_1 = w_{11} \cdot x_1 + w_{12} \cdot x_2 + w_{13} \cdot x_3$$

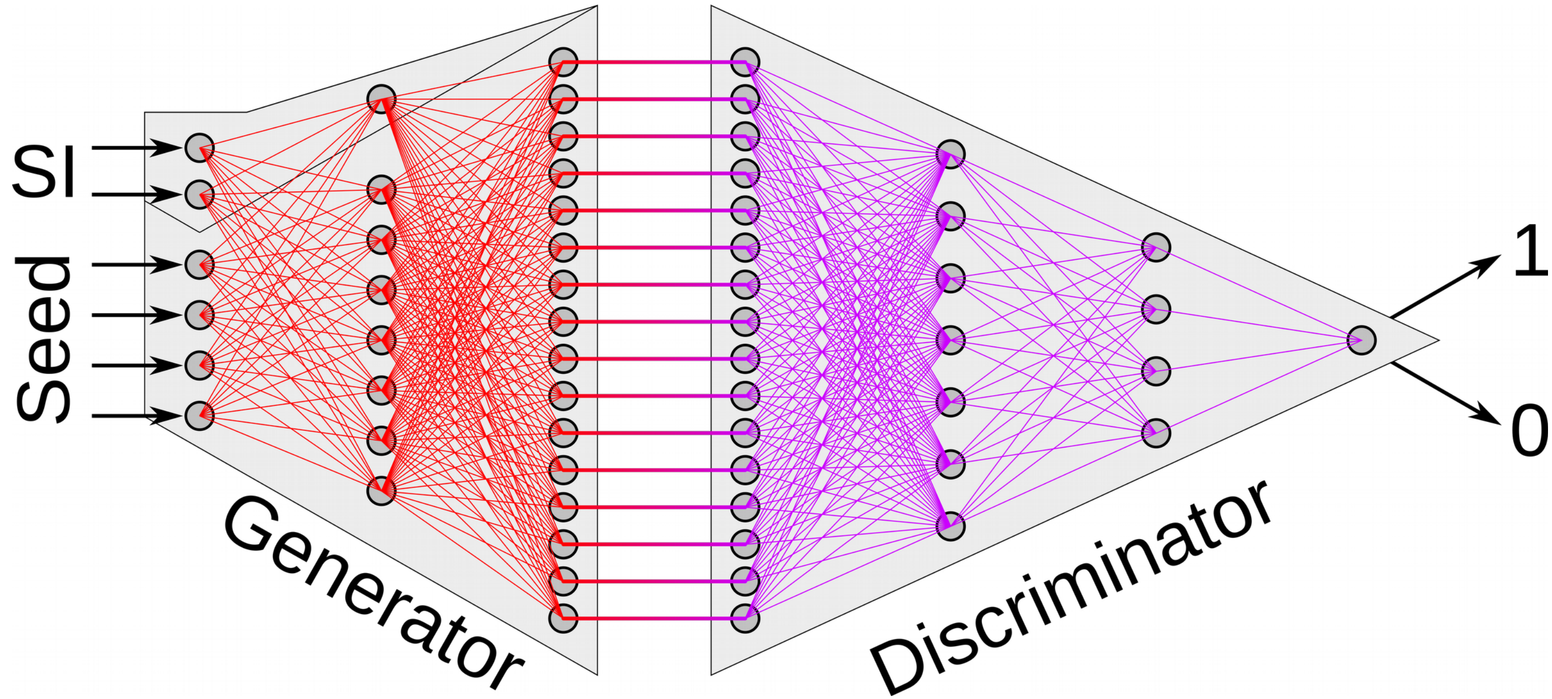
$$y_2 = w_{21} \cdot x_1 + w_{22} \cdot x_2 + w_{23} \cdot x_3$$

$$\vec{y} = w \cdot \vec{x} \ (+\vec{b})$$

$$\vec{f} = a(\vec{y})$$

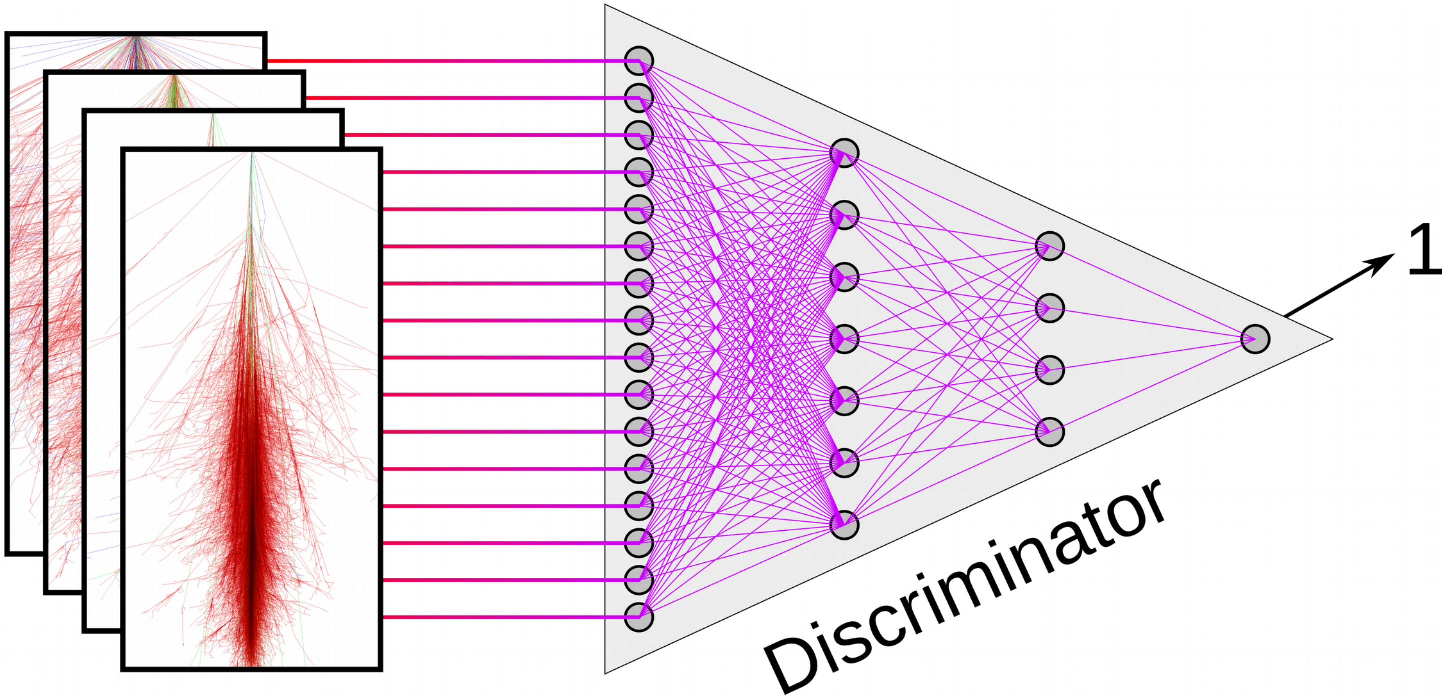
- Combination of linear and non-linear functions
- Training via loss function / metric on data pairs $(\vec{x},\ \vec{t})$
- $L = L(\vec{f}(\vec{x}),\ \vec{t}) \quad \Rightarrow \quad w' = w - \alpha \cdot \nabla_w L$

Generative Adversarial Neural Network (GAN)



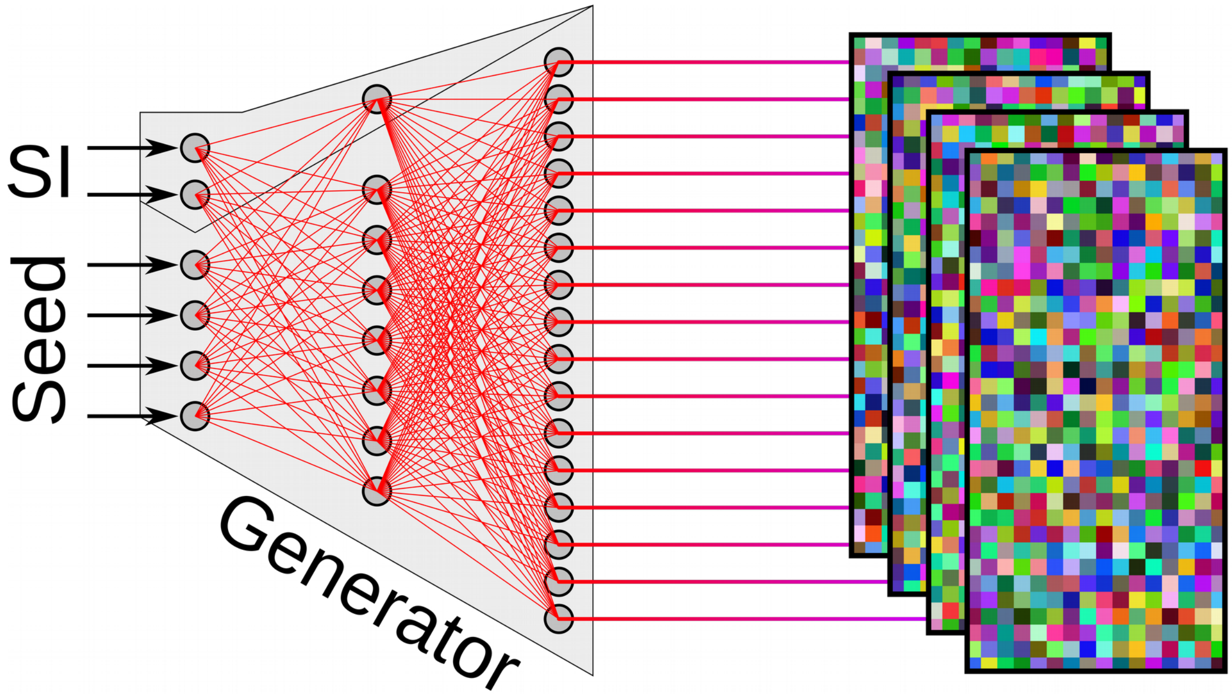
- Train discriminator on real (1) and generated (0) data
- Train generator to outsmart the discriminator

Training: Discriminator (Part 1)



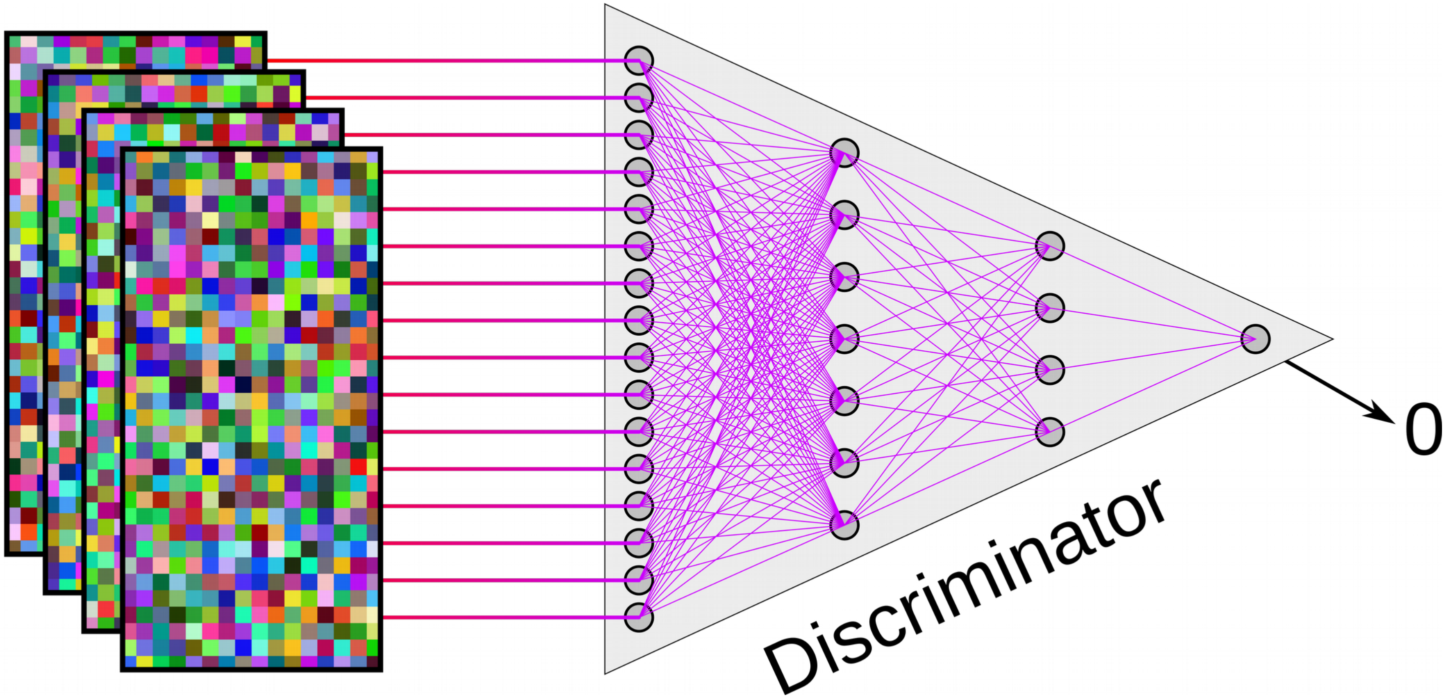
- Train discriminator on real (1) and generated (0) data
- Train generator to outsmart the discriminator

Training: Sampling



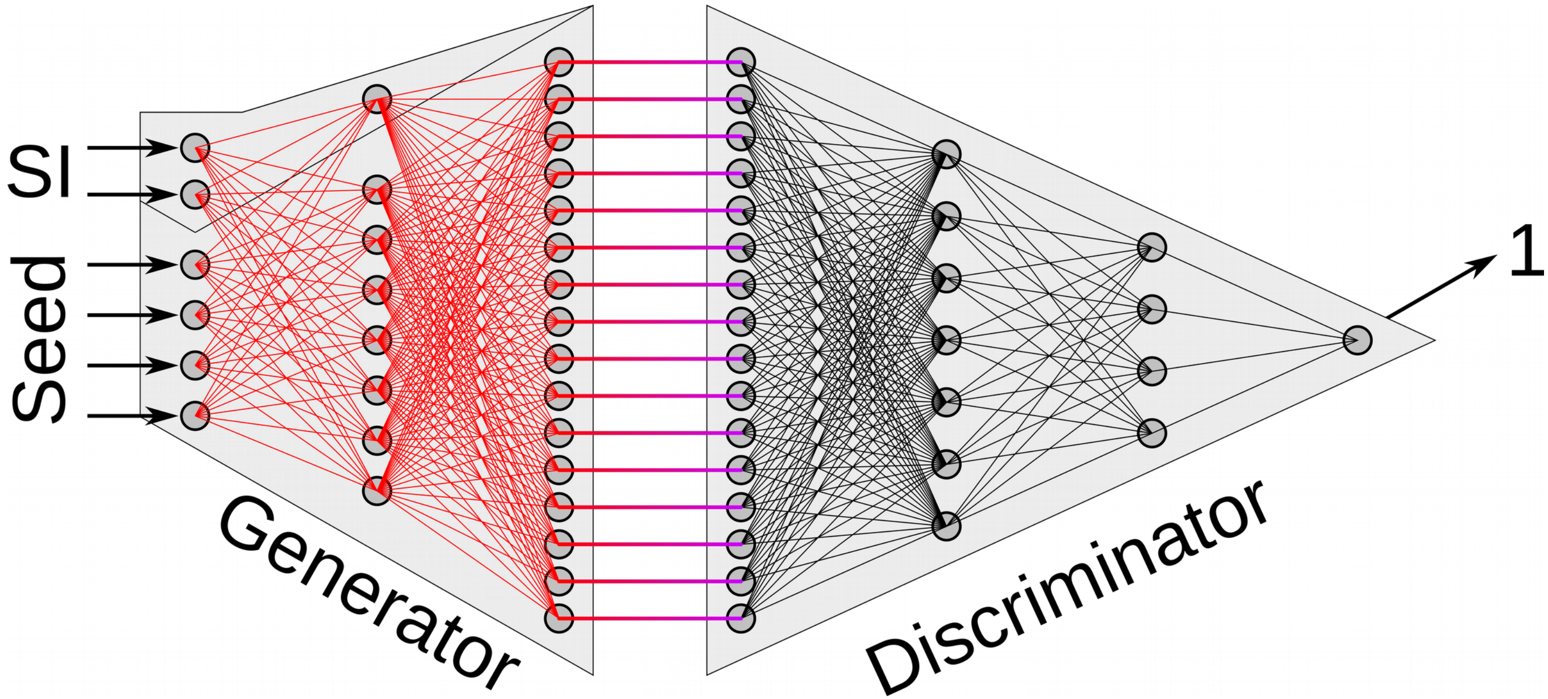
- Train discriminator on real (1) and generated (0) data
- Train generator to outsmart the discriminator

Training: Discriminator (Part 2)



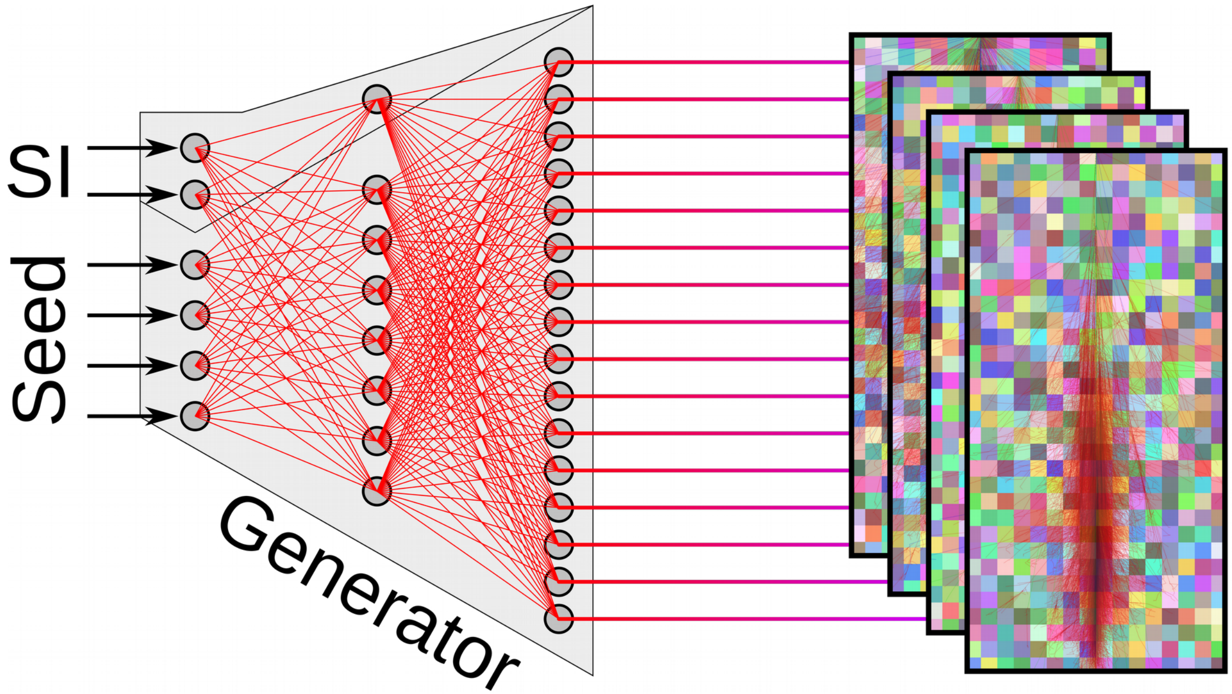
- Train discriminator on real (1) and generated (0) data
- Train generator to outsmart the discriminator

Training: Generator



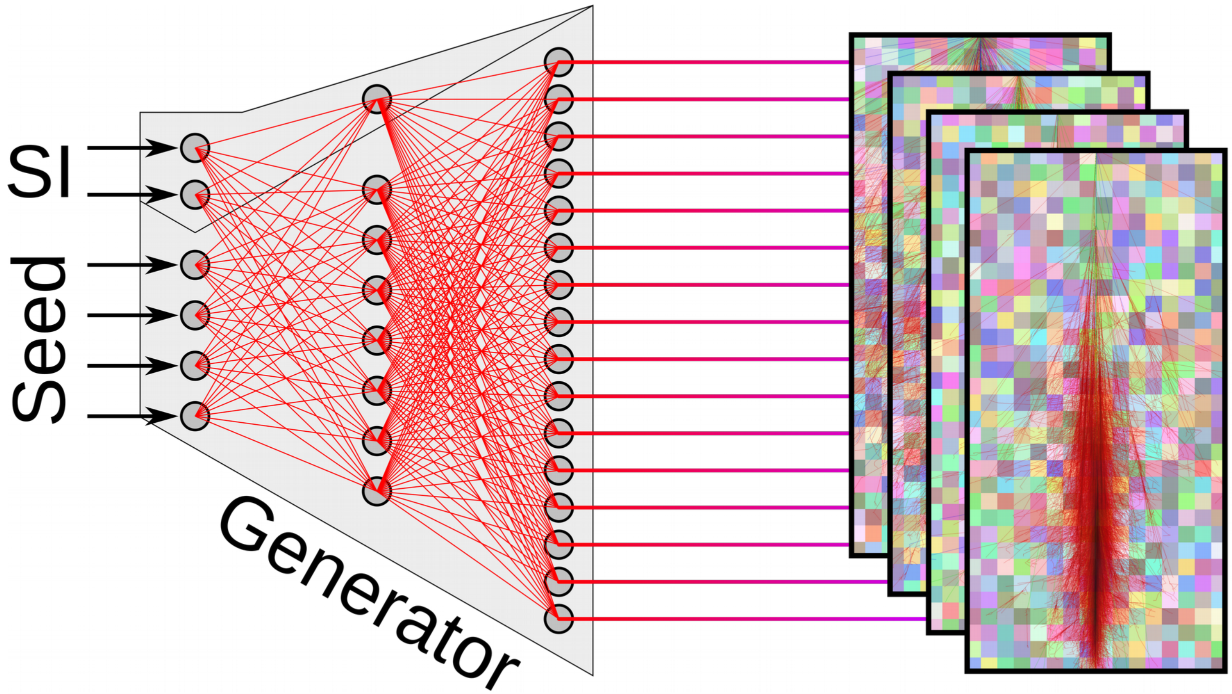
- Train discriminator on real (1) and generated (0) data
- Train generator to outsmart the discriminator

Training: Result



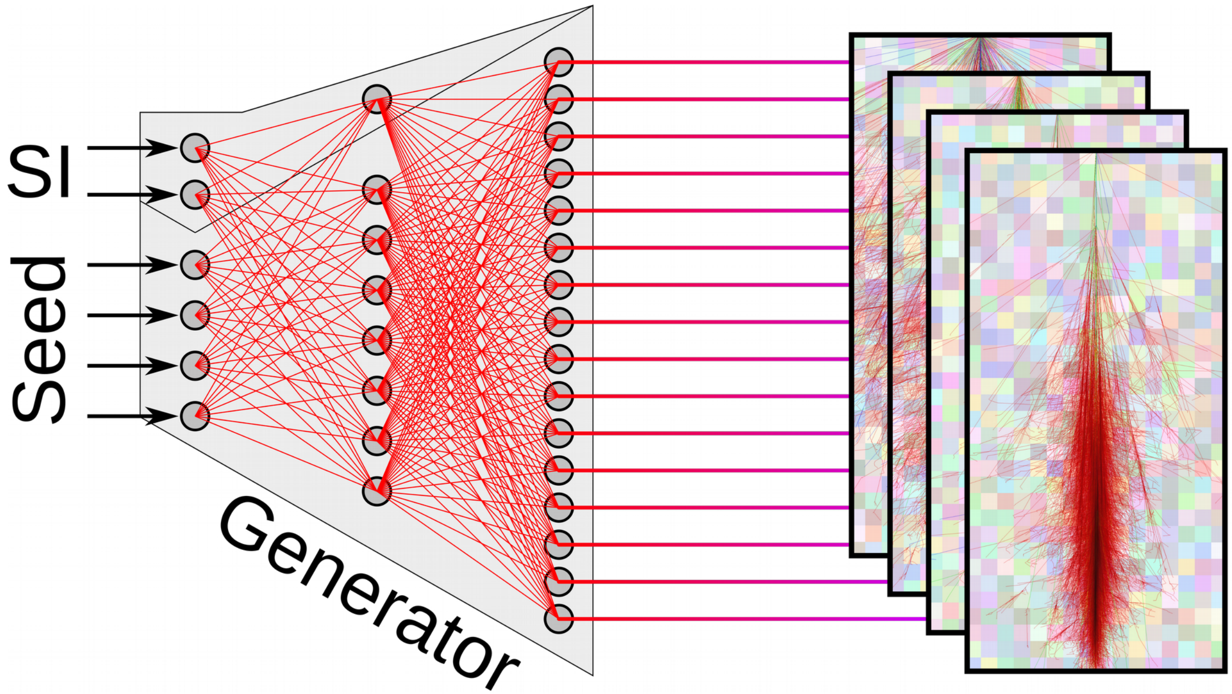
- Train discriminator on real (1) and generated (0) data
- Train generator to outsmart the discriminator

Training: Result



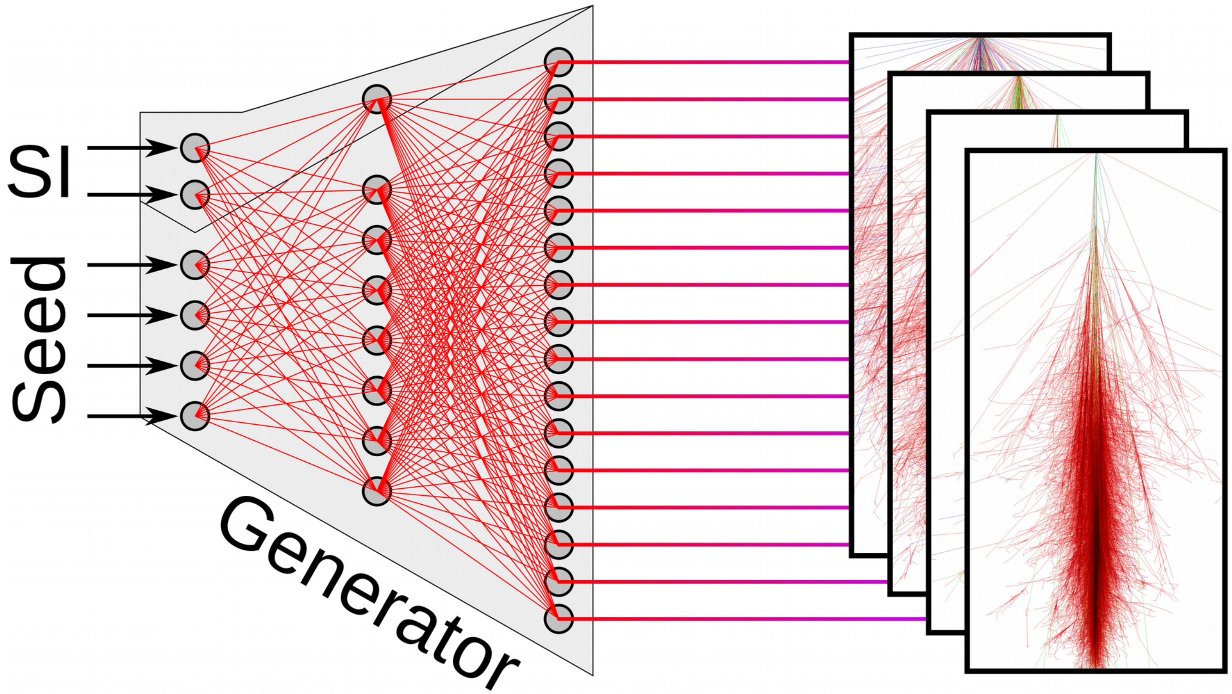
- Train discriminator on real (1) and generated (0) data
- Train generator to outsmart the discriminator

Training: Result



- Train discriminator on real (1) and generated (0) data
- Train generator to outsmart the discriminator

Training: Result



- Train discriminator on real (1) and generated (0) data
- Train generator to outsmart the discriminator

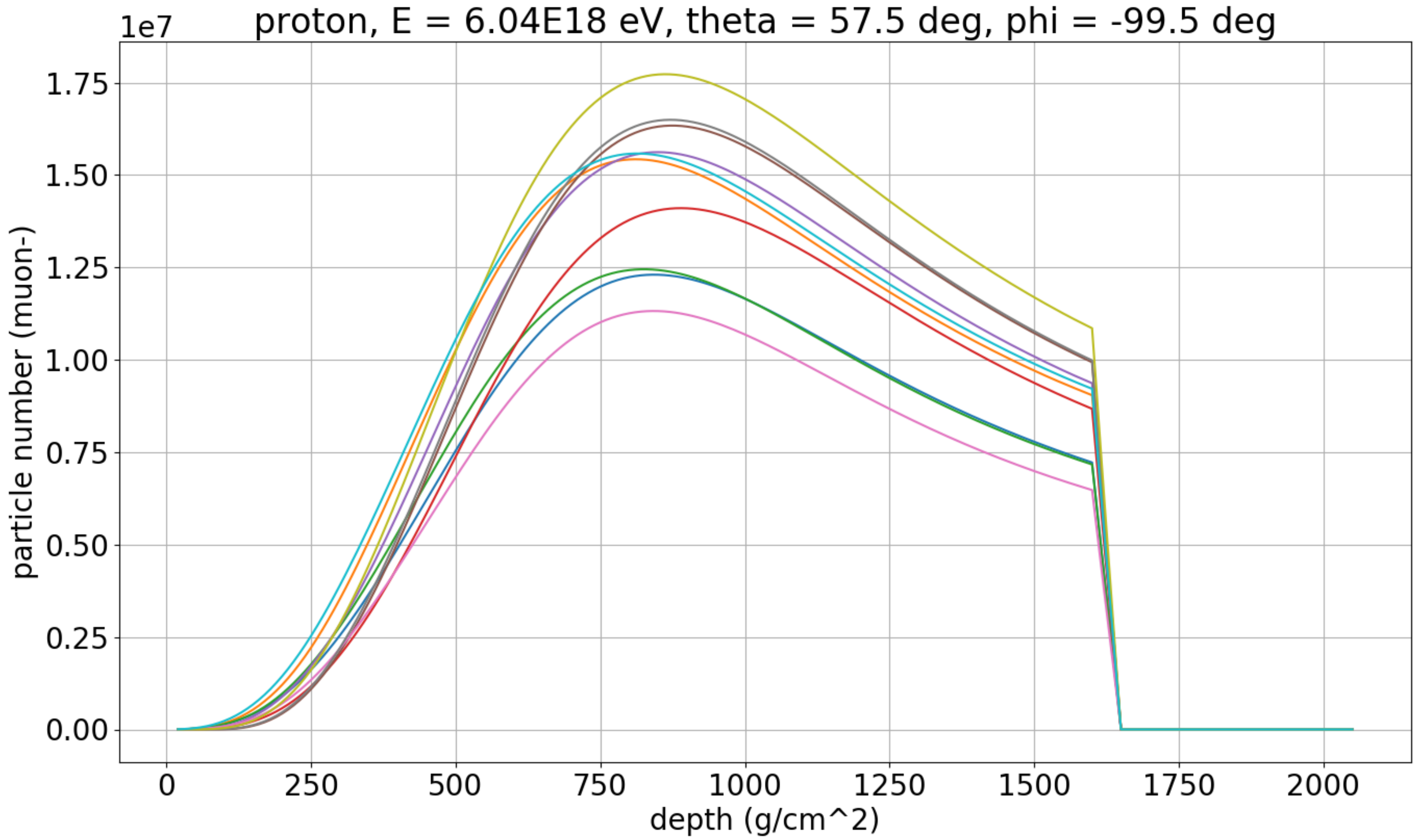
First Test (CONEX)

- CONEX: Hybrid Extenisve Air Shower Simulation
  - first: Monte Carlo until energy threshold (3D)
  - then: cascade equation solver (1D)
  - provides longitudinal profile only
  - runtime: seconds – minutes

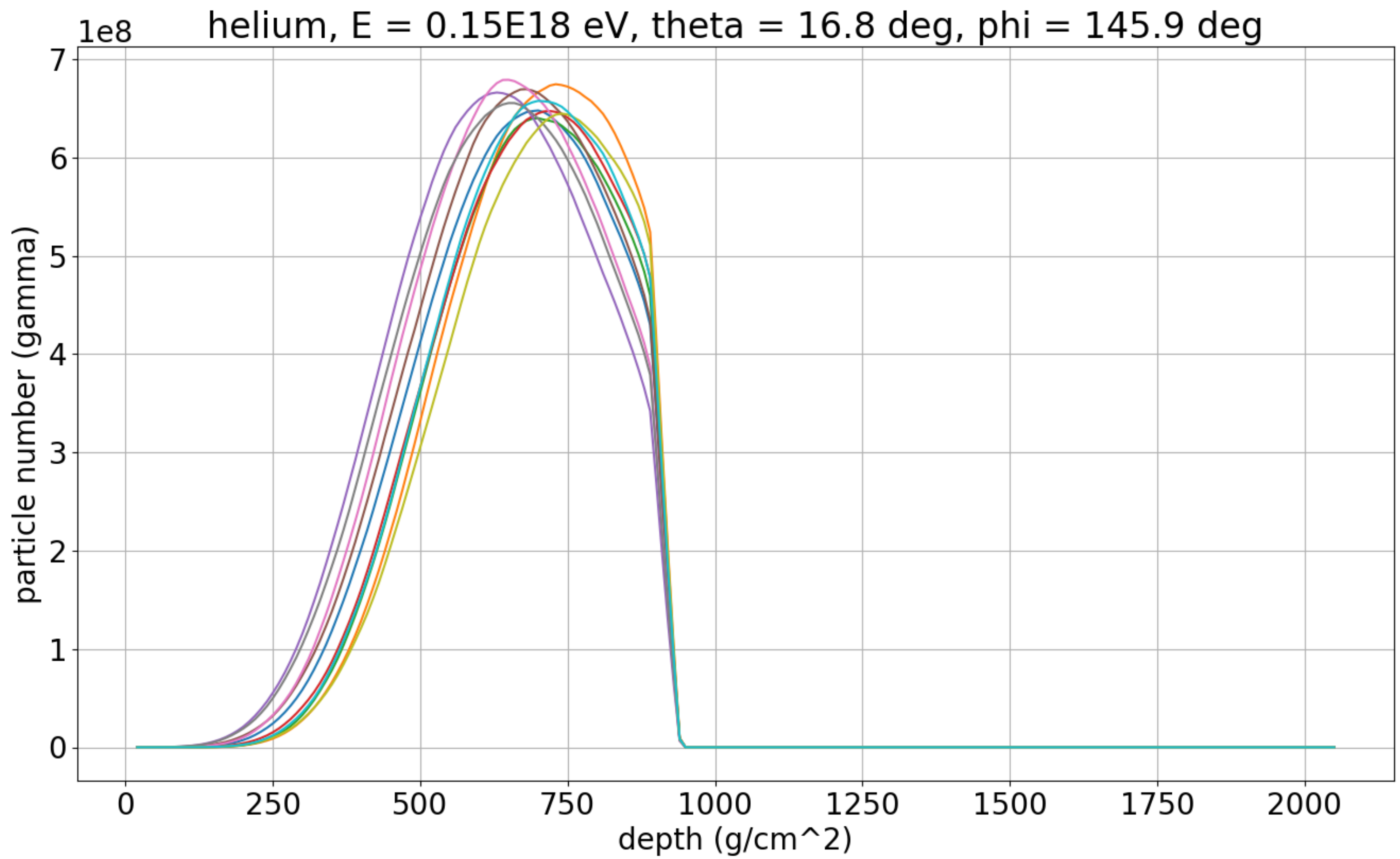
- Configuration:
  - E = 1E17 ... 1E19 eV
  - Zenith = 0 ... 65 deg
  - Azimuth = -180 ... 180 deg

- Generated ~187k datapoints

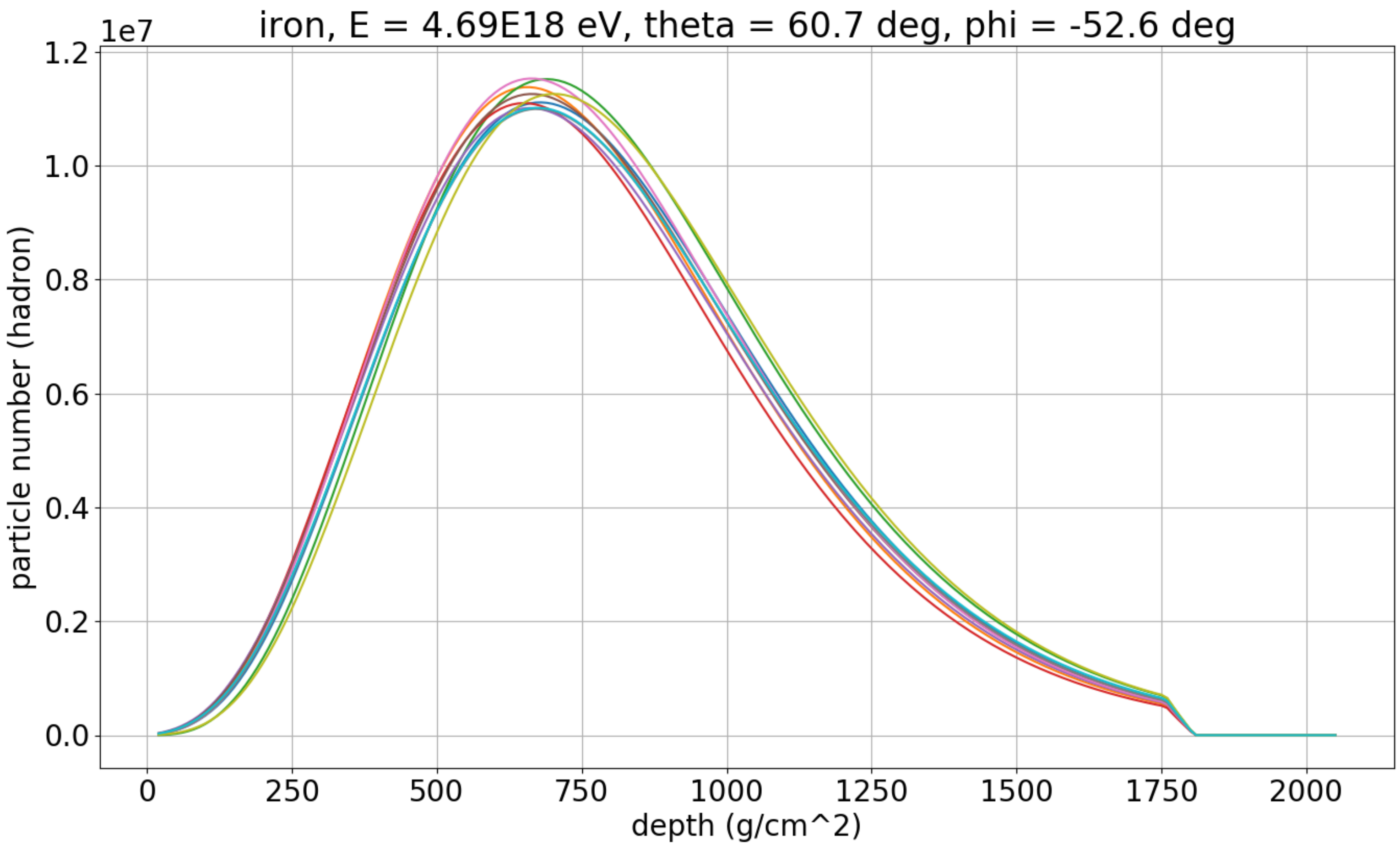
Shower-to-Shower Fluctuations



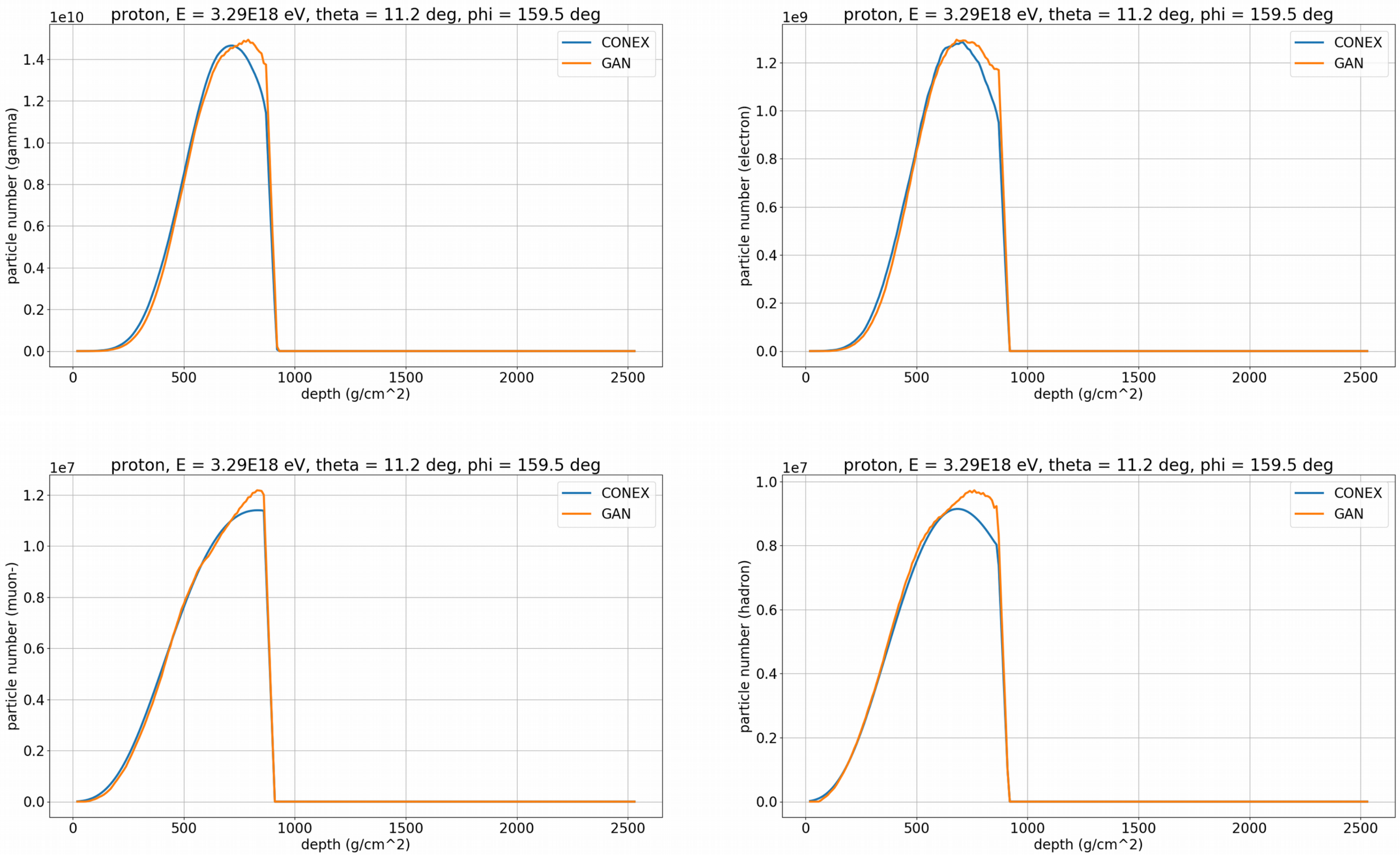
Shower-to-Shower Fluctuations



Shower-to-Shower Fluctuations

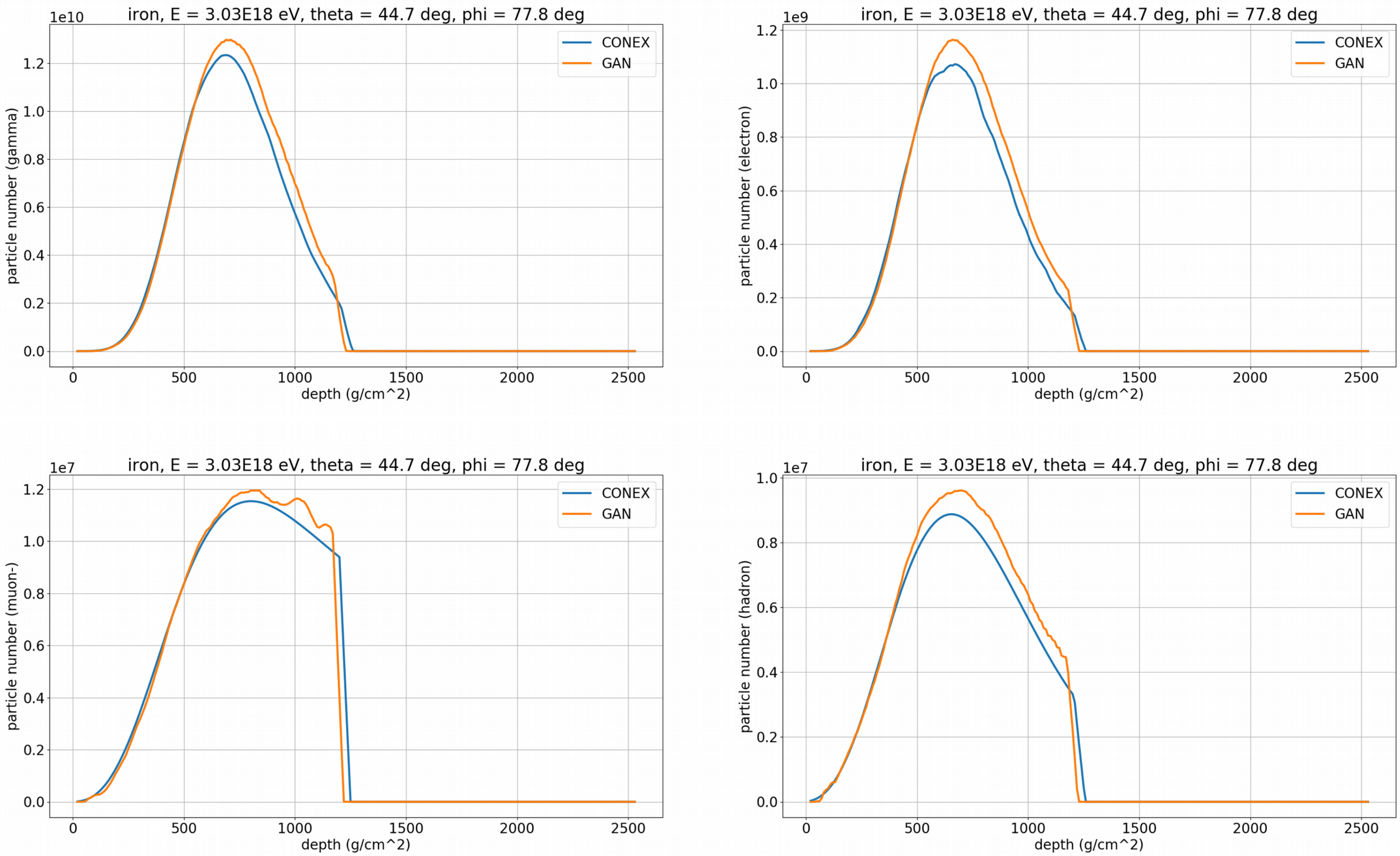


CONEX vs. GAN



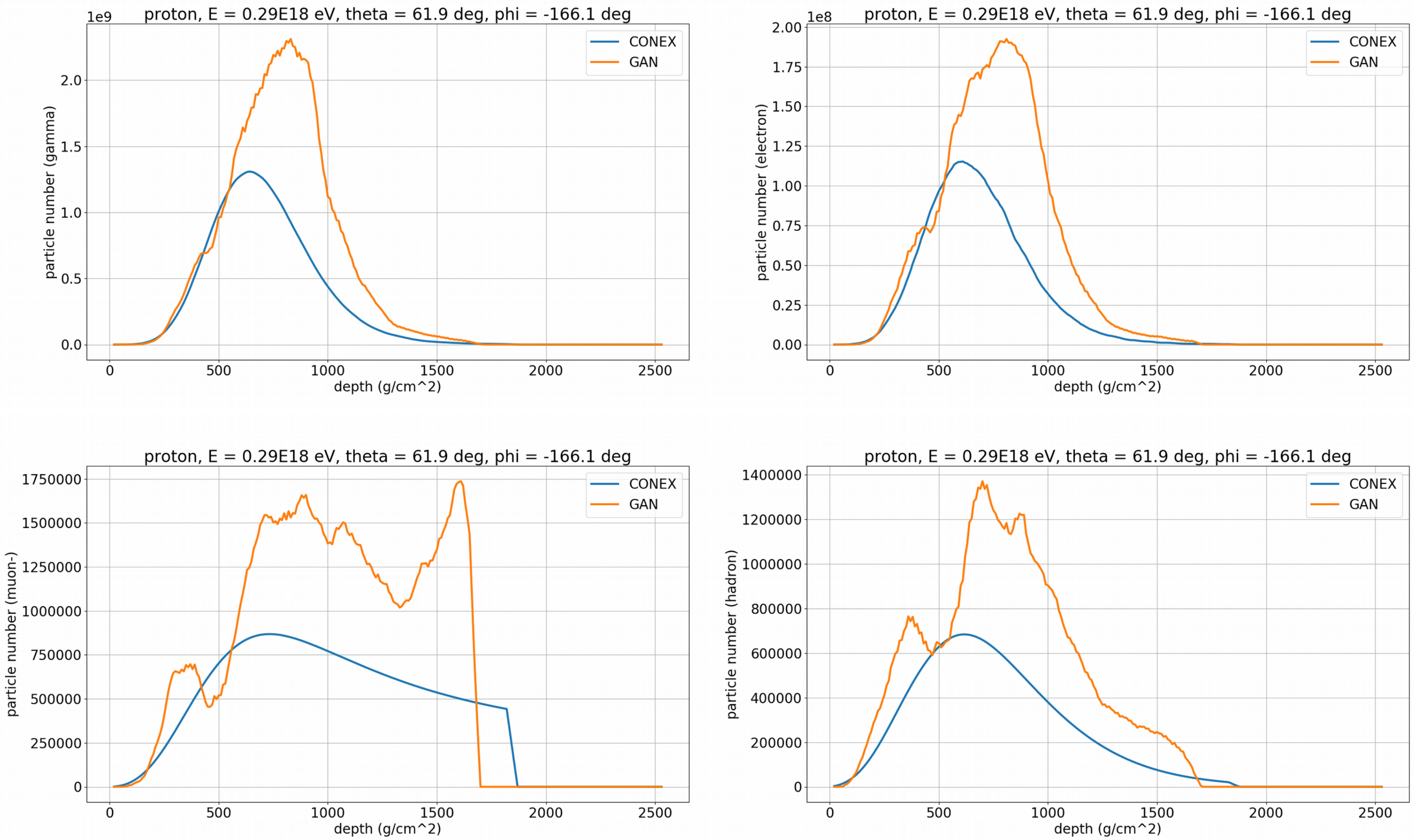
proton, E = 3.29E18 eV, theta = 11.2 deg, phi = 159.5 deg
1e10
particle number (gamma)
depth (g/cm^2)
CONEX
GAN
proton, E = 3.29E18 eV, theta = 11.2 deg, phi = 159.5 deg
1e9
particle number (electron)
depth (g/cm^2)
CONEX
GAN
proton, E = 3.29E18 eV, theta = 11.2 deg, phi = 159.5 deg
1e7
particle number (muon-)
depth (g/cm^2)
CONEX
GAN
proton, E = 3.29E18 eV, theta = 11.2 deg, phi = 159.5 deg
1e7
particle number (hadron)
depth (g/cm^2)
CONEX
GAN

CONEX vs. GAN

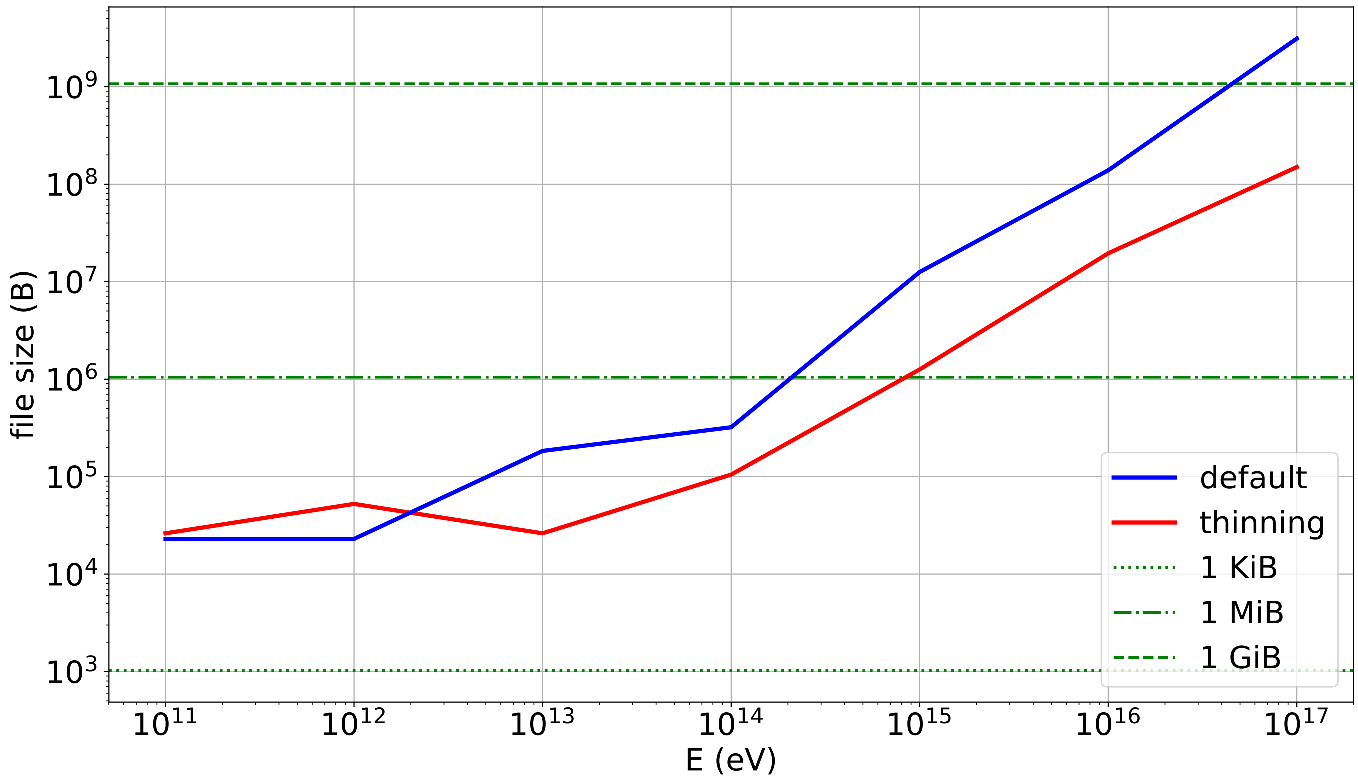


iron, E = 3.03E18 eV, theta = 44.7 deg, phi = 77.8 deg
1e10
particle number (gamma)
depth (g/cm^2)
CONEX
GAN
iron, E = 3.03E18 eV, theta = 44.7 deg, phi = 77.8 deg
1e9
particle number (electron)
depth (g/cm^2)
CONEX
GAN
iron, E = 3.03E18 eV, theta = 44.7 deg, phi = 77.8 deg
1e7
particle number (muon-)
depth (g/cm^2)
CONEX
GAN
iron, E = 3.03E18 eV, theta = 44.7 deg, phi = 77.8 deg
1e7
particle number (hadron)
depth (g/cm^2)
CONEX
GAN

CONEX vs. GAN


proton, E = 0.29E18 eV, theta = 61.9 deg, phi = -166.1 deg
1e9
particle number (gamma)
depth (g/cm^2)
CONEX
GAN
1e8
particle number (electron)
particle number (muon-)
particle number (hadron)

Shower Library



- Shower library required for analyses and model training
- Trained model = effective compression of shower library

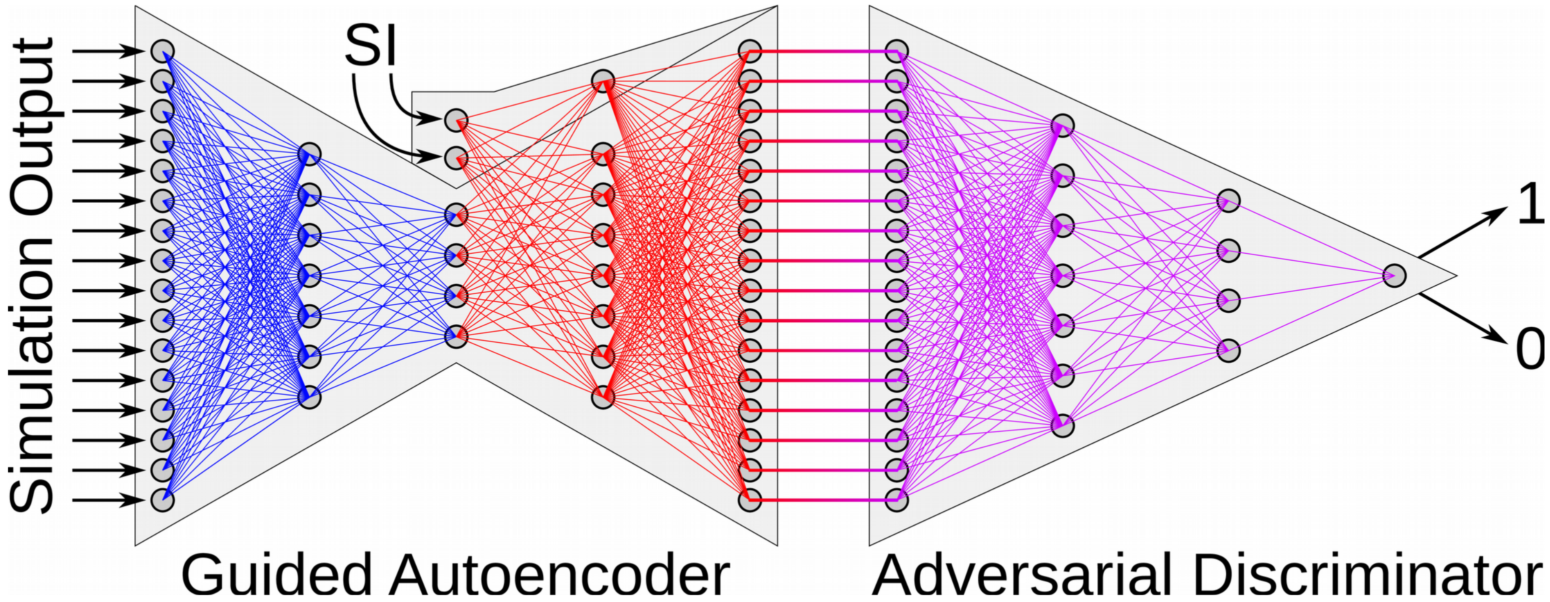
Summary

- Deep Neural Networks can serve as an effective simulation heuristic
- Can improve runtime and (effective) data compression
- Can employ specialized hardware
- Trained models can be conditioned on meta-parameters
- Adjustable accuracy is possible
- ErUM: Innovative Digitale Technologien zur Erforschung von Universum und Materie

Backup

Fast Implicit Simulation Heuristic (FISH)

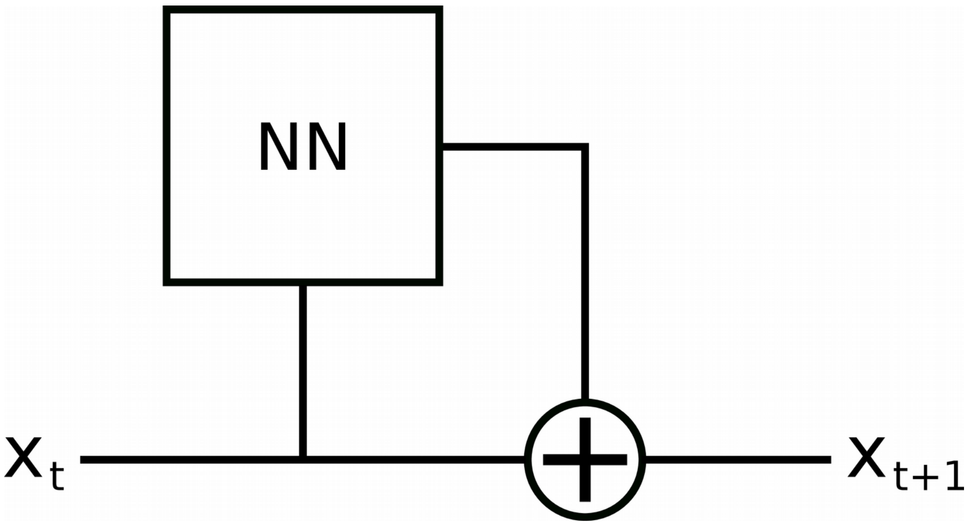
- Autoencoder with Adversarial Metric



- Simulation Input (SI) can be extended with meta-parameters
- Discriminator can be refined with real measurements

Adjustable Accuracy [2]

- ResNet



- Translate to ordinary differential equation (ODE)

$$x_{t+1} = x_t + f(x_t, \theta_t) \quad \Rightarrow \quad \frac{dx(t)}{dt} = f(x(t), t, \theta)$$

- Solve with standard ODE solver

- Adapt solver accuracy on the fly (training: high, inference: low)

Working Example [3]

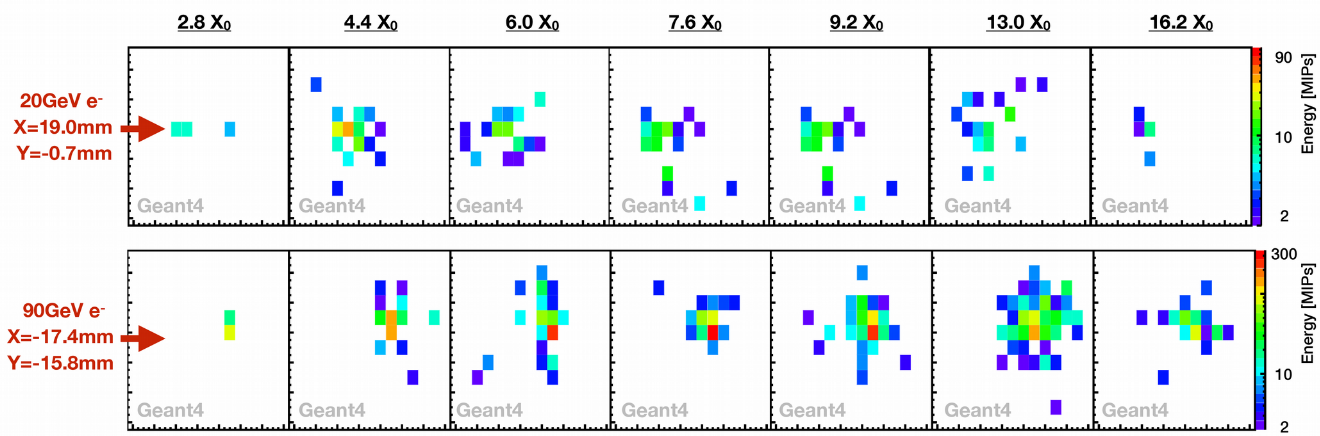
- „Precise simulation of electromagnetic calorimeter showers using a Wasserstein Generative Adversarial Network“ - Martin Erdmann, Jonas Glombitza, Thorben Quast – arXiv: 1807.01954

<u>Some Highlights:</u>

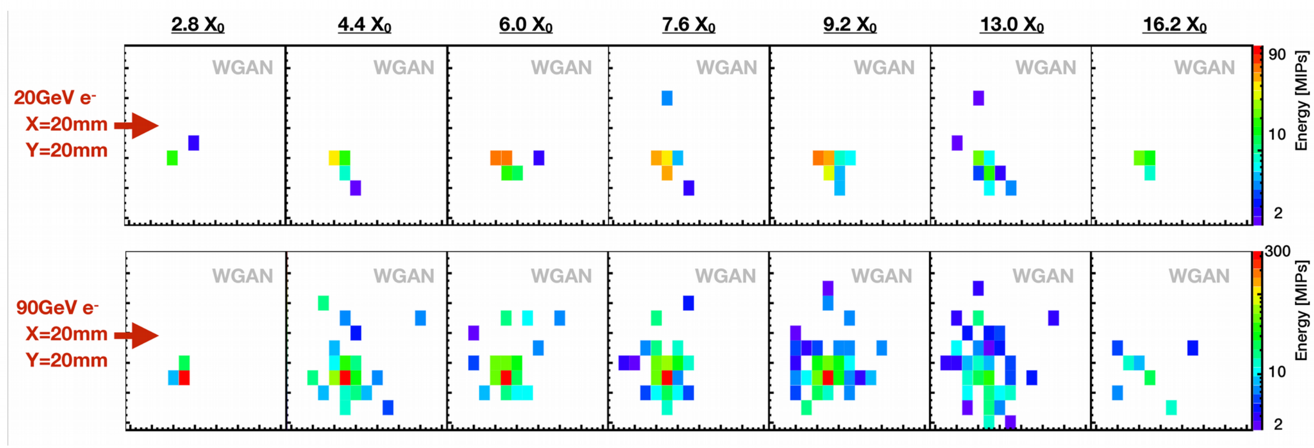
- **Wasserstein GAN** for HEP Geant4 simulations
- **Interpolation** between training energies
- Up to **x6660 faster** than Geant4
- Runtime is **independent of particle energy**

Geant4 vs. Wasserstein GAN [3]

- Geant4:



- Wasserstein GAN:



References

- Title picture:
  Karlsruhe Castle - Meph666 [CC BY-SA 3.0]
  https://commons.wikimedia.org/wiki/File:Karlsruhe-Schloss-meph666-2005-Apr-22.jpg

- Backup picture:
  Photo by Anthony from Pexels

- [1] CORSIKA 7: https://www.ikp.kit.edu/corsika/

- [2] „Neural Ordinary Differential Equations“ - Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, David Duvenaud – arXiv: 1806.07366

References

- [3] „Precise simulation of electromagnetic calorimeter showers using a Wasserstein Generative Adversarial Network“ - Martin Erdmann, Jonas Glombitza, Thorben Quast – arXiv: 1807.01954